In the Claims

Please amend the claims as shown below.

1. (Currently amended) A method, comprising:

generating a plurality of test designs, the plurality of test designs having varied characteristics to allow testing of a design automation tool, wherein generating one of the plurality of test designs comprises:

instantiating an input/output (I/O) structure of a top level module, the top level module having input and output pins;

selecting a plurality of submodules from a design module library, wherein a probabilistic function is applied to select the plurality of submodules of different types from the library, wherein said selecting the plurality of submodules is constrained based on a hardware family of the one of the test designs, wherein the hardware family is selected from a plurality of hardware families;

parameterizing the plurality of submodules from the design module library for interconnection with the top level module, the plurality of submodules having input and output lines;

providing logic to interconnect the plurality of parameterized submodules as well as to connect the plurality of parameterized submodules to various input and output pins of the top level module;

determining whether a predetermined number of the test designs for testing the design automation tool has been generated;

applying the plurality of test designs to test the design automation tool.

2. (Previously presented) The method of claim 1, wherein the design automation tool is used to implement hardware descriptor language designs on a programmable chip.

3. (Previously presented) The method of claim 1, wherein the plurality of submodules comprise a memory module and a Digital Signal Processor (DSP) core.

4. (Previously presented) The method of claim 1, wherein instantiation constraints are used to select the plurality of submodules.

5. (Previously presented) The method of claim 1, wherein the design automation tool is a synthesis or a place and route tool.

6. (Previously presented) The method of claim 1, wherein providing logic to interconnect the plurality of parameterized modules comprises identifying inputs and outputs.

7. (Previously presented) The method of claim 6, wherein inputs comprise input pins of the top level module, submodule output lines, and registers.

8. (Previously presented) The method of claim 6, wherein outputs comprise output pins of the top level module, submodule input lines, and registers.

9. (Previously presented) The method of claim 8, wherein providing logic to interconnect the plurality of parameterized submodules further comprises classifying inputs and outputs as clock lines, control lines, and data lines.

10. (Previously presented) The method of claim 8, wherein generating one of the plurality of test designs further comprises:

generating randomized logic.

11. (Previously presented The method of claim 10, wherein randomized logic is generated to drive outputs.

12. (Previously presented) The method of claim 10, wherein generating randomized logic comprises directly wiring outputs to inputs, generating a logic expression using inputs, generating a mathematical expression using inputs, or generating decision logic.

13. (Previously presented) The method of claim 6, wherein parameterizing the plurality of submodules comprises defining interfaces, data width, and the type of signal for input and output lines associated with the submodule.

14. (Previously presented) The method of claim 6, wherein submodules comprise adders, phase lock loops, memory, and timers.

15. (Previously presented) The method of claim 6, wherein generating one of the plurality of test design further comprises selecting a clock structure for each output.

16. (Previously presented) The method of claim 15, wherein clock structures include a plurality of synchronous and asynchronous structures.

17. (Currently amended) A computer system, comprising:

memory operable to hold information associated with a design module library;

a processor coupled to memory, the processor configured to generate a plurality of test designs, the plurality of test designs having varied characteristics to allow testing of a design automation tool, wherein generating one of the plurality of test designs comprises:

instantiating an input/output (I/O) structure of a top level module, the top level module having input and output pins;

selecting a plurality of submodules from the design module library, wherein the plurality of submodules of different types are randomly selected from the library, wherein said selecting the plurality of submodules is constrained based on a hardware family of the one of the test designs, wherein the hardware family is selected from a plurality of hardware families;

parameterizing the plurality of submodules from the design module library for interconnection with the top level module, the plurality of submodules having input and output lines;

providing logic to interconnect the plurality of parameterized submodules as well as to connect the plurality of parameterized submodules to various input and output pins of the top level module[[,]];

wherein the processor is configured to determine whether a predetermined number of the test designs for testing the design automation tool has been generated,

wherein the plurality of test designs are applied to test the design automation tool.

18. (Original) The computer system of claim 17, wherein the design automation tool is used to implement hardware descriptor language designs on a programmable chip.

19. (Original) The computer system of claim 17, wherein the design automation tool is used to implement designs on an ASIC.

20. (Original) The computer system of claim 17, wherein the design automation tool is an electronic design automation tool.

21. (Original) The computer system of claim 17, wherein the design automation tool is a synthesis or a place and route tool.

22. (Previously presented) The computer system of claim 17, wherein providing logic to interconnect the plurality of parameterized submodules comprises identifying inputs and outputs.

23. (Original) The computer system of claim 22, wherein inputs comprise input pins of the top level module, submodule output lines, and registers.

24. (Original) The computer system of claim 22, wherein outputs comprise output pins of the top level module, submodule input lines, and registers.

25. (Currently amended) An apparatus for generating test a testbench, the apparatus comprising:

storage means for storing data;

processing means for generating a plurality of test designs, the plurality of test designs having varied characteristics to allow testing of a design automation tool, wherein said processing means is coupled to said storage means, said processing means ~~instantiates~~ for instantiating an

input/output (I/O) structure of a top level module, the top level module having input and output pins, said processing means ~~selects~~ for selecting a plurality of submodules from a design module library, wherein a probabilistic function is applied to select the plurality of submodules of different types from the library and the selection of the plurality of submodules is constrained based on a hardware family of one of the test designs, the hardware family is selected from a plurality of hardware families, said processing means ~~parameterizes~~ for parameterizing the plurality of submodules from the design module library for interconnection with the top level module, the plurality of submodules having input and output lines, [[and]] said processing means ~~provides~~ for providing logic to interconnect the plurality of parameterized submodules as well as to connect the plurality of parameterized submodules to various input and output pins of the top level module, wherein the processing means is further configured to determine whether a predetermined number of the test designs for testing the design automation tool has been generated, wherein the processing means is further configured to repeat selecting a plurality of submodules from a design module library upon determining that the predetermined number of test designs is not generated.

26. (Previously presented) The apparatus of claim 25, wherein the design automation tool is used to implement hardware descriptor language designs on a programmable chip.

27. (Previously presented) The apparatus claim 25, wherein the design automation tool is used to implement designs on an ASIC.

28. (New) The method of claim 1, further comprising selecting a plurality of submodules upon said determining that the predetermined number of the test designs is not generated.

29. (New) The method of claim 1, wherein the predetermined number is provided by a user.

30. (New) The computer system of claim 17, wherein the processor is configured to select a plurality of submodules upon said determining that the predetermined number of the test designs is not generated.

31. (New) The computer system of claim 17, wherein the predetermined number is provided by a user.

32. (New) The apparatus of claim 25, wherein the predetermined number is provided by a user.